

(19) Japan Patent Office (JP)

(12) Publication of Unexamined Patent Application (A)

(11) Japanese Patent Laid-Open Number: Tokkaihei 10-49386

(43) Laid-Open Date: Heisei 10-2-20 (February 20, 1998)

(51) Int.Cl. ⁶	Identification Code	Office Reference Number	FI
G06F 9/46	340	G06F 9/46	340F

Technology Manifestation Part

Request for Examination: Done

Number of Claims: 4 OL (9 pages in total)

(21) Application Number: Tokuganhei 8-204596

(22) Filed: Heisei 8-8-2 (August 2, 1996)

(71) Applicant: 000004237

Nippon Electric Corporation
5-7-1 Shiba, Minatoku, Tokyo

(72) Inventor: Ono Kazuki

Nippon Electric Corporation
5-7-1 Shiba, Minatoku, Tokyo

(74) Agent: Patent Attorney Gotou Yousuke (and two agents)

(54) [Title of the Invention] RESOURCE EXCLUSION SYSTEM

(57) [Abstract]

[Object] To provide a resource exclusion system in which a series of operations necessary for exclusive control are shared by each process to eliminate a bottleneck in the process of an exclusive control operation thereby preventing occurrence of a reduction in throughput even when the multiplicity of the process is increased.

[Solving Means] Each of processes 13a and 13b intending to hold a resource makes reference to and updates an exclusive control management table 12 that includes a resource name and an exclusion queue. When the processes 13a and 13b intend to hold a resource, each of them registers its process identifier at the tail end of the exclusion queue when process identifiers are registered thereon. Whereas when such process identifiers are not registered on the exclusion queue, each process registers its process identifier at the head of the exclusion queue and holds the resource. When releasing the resource, each process deletes its process identifier registered on the exclusion queue. At this point, each process issues resource exclusion termination notification if process identifiers of other processes are registered on the exclusion queue.

[Scope of Claim]

[Claim 1] A resource exclusion system in which a resource, which may be requested by a plurality of processes for use, is exclusively used by any one of the plurality of processes,

wherein each of the plurality of processes makes reference to and updates an exclusive control management table including the name of the resource and an exclusion queue corresponding to the name of the resource name.

[Claim 2] The resource exclusion system according to Claim 1,

wherein each of the plurality of processes makes reference to the exclusive control management table at the moment when they intend to hold

the resource, and when any one of process identifiers that are provided to the plurality of processes is registered on the exclusion queue, each process registers its process identifier at the tail end of the exclusion queue and waits until exclusive use of the resource ends, whereas when the identifiers are not registered on the exclusion queue, each process registers its identifier at the head of the exclusion queue and holds the resource.

[Claim 3] The resource exclusion system according to Claim 2,

wherein each of the plurality of processes deletes its identifier registered on the exclusion queue and releases the held resource at the moment when they intend to release the resource, and notifies a process corresponding to the identifier of release of the resource when an identifier is registered on the exclusion queue item corresponding to the released resource.

[Claim 4] The resource exclusion system according to any one of Claims 1 to 3,

wherein the exclusion queue of the exclusive control management table is created and deleted by the plurality of processes.

[Detailed Description of the Invention]

[0001]

[Technical Field to which the Invention Pertains]

The present invention relates to a resource exclusion system, more specifically to a resource exclusion system for managing resource exclusion operations among a plurality of processes in a computer system.

[0002]

[Prior Art]

In a conventional resource exclusion system for performing exclusive operations on resources among a plurality of processes, the process that performs an exclusive control operation performs all of the following operations as requested by the process that has requested access to a resource: receipt of an exclusion request, registration of exclusion

information to the exclusion management table, locking of the resource until it is released from the process by which it has been requested, receipt of request of resource release, deletion of the exclusion information from the exclusion management table upon releasing of the resource, and notification of release of the lock upon releasing of the resource. For example, Japanese Patent Laid-Open No.Hei05-113891 and Japanese Patent Laid-Open No.Hei05-20269 propose a variety of systems for this kind of resource exclusion system.

[0003]

[Problems to be Solved by the Invention]

The first problem is that the process of exclusive control operations for resource exclusion is a bottleneck in the entire system.

[0004]

This is because the process that performs an exclusive control operation is required to perform it for all of the other processes requesting access to a resource. In other words, the process that performs an exclusive control operation is required to perform a series of exclusive control operations such as receipt of an exclusion request, registration of exclusion information to the exclusion management table, locking of the resource until it is released from the process by which it has been requested, receipt of request of resource release, deletion of the exclusion information from the exclusion management table upon releasing of the resource, and notification of release of the lock upon releasing of the resource, for each of the other processes requesting access to a resource. As the number of processes requiring an exclusive control operation increases so too does the overhead of the process of the exclusive control operation. This becomes a bottleneck in the system processing, resulting in the reduction in total process (entire system) throughput.

[0005]

It is an object of the present invention to provide a resource exclusion

system in which a series of operations required for exclusive control are shared by each process to eliminate a bottleneck in the process of an exclusive control operation and thereby preventing occurrence of a reduction in throughput even when the multiplicity of the process is increased.

[0006]

[Means for Solving the Problems]

According to the present invention, it is possible to provide a resource exclusion system in which a resource, which may be requested by a plurality of processes to be held, is exclusively used by any one of the plurality of processes. The resource exclusion system is characterized in that each of the plurality of processes makes reference to and updates an exclusive control management table including the name of the resource and an exclusion queue corresponding to the name of the resource.

[0007]

Here, each of the plurality of processes makes reference to the exclusive control management table at the moment when they intend to hold the resource, and when any one of process identifiers that are provided to the plurality of processes is registered on the exclusion queue, each process registers its identifier at the tail end of the exclusion queue and waits until exclusive use of the resource ends. Whereas when any one of the identifiers is not registered on the exclusion queue, each process registers its process identifier at the head of the exclusion queue and holds the resource.

[0008]

In addition, each of the plurality of processes deletes its identifier registered on the exclusion queue and releases the held resource at the moment when they intend to release the resource, and notifies a process corresponding to the identifier of release of the resource when an identifier is registered on the exclusion queue corresponding to the released resource.

[0009]

[Operations]

The exclusive control management table includes the name of a resource to be used exclusively, and an exclusion queue corresponding to the name of the resource. A process makes reference to the exclusive control management table to hold a resource, determines that none of the identifiers of other processes is registered for the resource which the process intends to hold, registers its identifier on the exclusion queue of the exclusive control management table, and holds the resource.

[0010]

When identifiers of other processes are registered on the exclusion queue of the resource to be held, the process registers its identifier at the tail end of the exclusion queue and waits until exclusive use of the resource ends.

[0011]

When performing a resource releasing operation, the process deletes its identifier from the exclusion queue of the target resource, which is provided on the exclusive control management table. In presence of identifiers of other processes that are registered on the exclusion queue, these processes are notified that exclusive use of the resource has ended.

[0012]

[Embodiments]

Hereinafter, the embodiments of the present invention will be described with reference to the drawings.

[0013]

Fig. 1 shows a first embodiment of a resource exclusion system of the present invention. The resource exclusion system includes: an exclusive control management table 12 that has resource names for identifying resources 11, and queues that correspond to each of the resources 11; and processes 13a and 13b, each of which can refer to and update the exclusive control management table 12. It should be noted that it is desirable that the exclusive control management table 12 be provided on a fast-reference device.

[0014]

Hereinafter, the operation of the resource exclusion system shown in Fig. 1 will be described with reference to Figs. 2 and 3 as well as Fig. 1.

[0015]

Referring to Fig. 2, the operation of holding a resource will be first described. Here, it is assumed that as the name of the resource 11 "resource A" is registered on the exclusive control management table 12 and that the queue corresponding to the resource A has no registered identifiers. When the process 13a intends to hold the resource 11 in this situation, it first makes reference to the exclusive control management table 12 in Step S21, and then determines in Step S22 the presence of identifiers registered on the exclusion queue of the target resource (the resource A, in this case). In Step S23B, the process 13a then registers its process identifier "process A" at the head (process identifier 1) of the exclusion queue of the target resource, and holds the target resource 11.

[0016]

When the process 13b intends to hold the resource 11 after this operation, since it will be determined in Step S22 that a process identifier is registered on the exclusion queue of the target resource 11, the process 13b registers its process identifier "process B" at the tail end (process identifier 2, in this case) of the exclusion queue of the target resource 11 in Step S23A. Then, in Step S24A, the process 13b shifts to a state where it waits for termination of exclusive use of the resource, and waits for resource exclusion termination notification from the process 13a holding the target resource 11.

[0017]

Next, the operation of releasing a resource will be described with reference to Fig. 3. Here, it is assumed that the exclusive control management table 12 is in the state shown in Fig. 1, that is, the process 13a holds the resource 11 and the process 13b is in a state where it waits for termination of exclusive use of a resource.

[0018]

In Step S31, the process 13a first makes reference to the exclusion queue of the held resource 11 in the exclusive control management table 12, and determines in Step S32 the presence of process identifiers waiting for the exclusive use of the processes on the exclusion queue of the target resource. Since the process identifier of the process 13b is registered on the exclusion queue shown in Fig. 1, the process 13a deletes its process identifier in Step S33A, allowing these process identifiers to move by one column toward the head. As a result, the process identifier "process B" is registered on the column of the process identifier 1. The process 13a then issues resource exclusion termination notification to the process 13b, which corresponds to the process identifier registered at the head of the exclusion queue. Subsequently, the process 13b that has received the resource exclusion termination notification holds the resource 11.

[0019]

When the process 13b intends to release the resource 11 after this operation, since it will be determined in Step S32 that the process identifiers waiting for the exclusive use of the processes are not registered on the exclusion queue of the target resource 11, the process 13b deletes its identifier in Step S33B, and then the resource releasing operation is completed.

[0020]

Next, a second embodiment of the present invention will be described with reference to Figs. 4 and 5. In this embodiment the exclusive control management table has no queues; each process creates a queue therein according to need.

[0021]

In the operation of holding a resource, as shown in Fig. 4, the process intending to hold a resource first makes reference to the exclusive control management table in Step S41, and then determines in Step S42 whether or

not the exclusion queue of the target resource is created in the exclusive control management table. When it is determined that the exclusion queue of the target resource is created in the exclusive control management table, the target resource is being held (exclusively used) by another process. For this reason, the process registers its process identifier at the tail end of the exclusion queue of the target resource in Step S43A. In Step S44, the process shifts to a state where it waits for termination of exclusive use of the resource, and waits for resource exclusion termination notification from the precedent process holding the target resource.

[0022]

If it is determined in Step S42 that no exclusion queue for the target resource is created in the exclusive control management table, the process is capable of exclusively using (holding) the target resource. Thus, the process creates an exclusion queue of the target resource and registers its process identifier at the head of that queue. In this way the process holds the target resource.

[0023]

Next, the operation of releasing a resource will be described with reference to Fig. 5. In Step S51, the process for releasing a resource first makes reference to the exclusion queue of the resource it has held, and then determines in Step S52 whether or not process identifiers of other processes are registered on the exclusion queue of the target resource. When it is determined that such process identifiers are registered on the exclusion queue of the target resource, processes that are waiting for termination of exclusive use of the target resource are present. For this reason, the process deletes its process identifier from the exclusion queue of the target resource in Step S53A, and in Step S54, each of the other process identifiers registered on the exclusion queue of the target resource is allowed to move by one column toward its head. Then, resource exclusion termination notification is issued to the process that corresponds to the process identifier

positioned at the head of the exclusion queue of the target resource, thereby allowing the process, which had waited for the termination of exclusive use of the resource, to use that resource.

[0024]

WHen it is determined in Step S52 that other process identifiers are not registered on the exclusion queue of the target resource, since no process is waiting for release of the target resource, the process deletes its process identifier, as well as the exclusion queue of the resource on the exclusive control management table, which will be released for use.

[0025]

[Effect of the Invention]

The first effect is that it is possible to avoid a reduction in throughput due to a bottleneck in the process of the exclusive control operation, even when the multiplicity of the process is increased.

[0026]

This is because a series of exclusive control operations such as receipt of an exclusion request, registration of exclusion information to the exclusion management table, locking of the resource until it is released from the process from which it has been requested, receipt of request of resource release, deletion of the exclusion information from the exclusion management table upon releasing of the resource, and notification of release of the lock upon releasing of the resource, are shared by processes which use the resource.

[Brief Description of the Invention]

Fig. 1 is a block diagram showing a first embodiment of the present invention.

Fig. 2 is a flowchart for explaining the resource-holding operation in the resource exclusion system shown in Fig. 1.

Fig. 3 is a flowchart for explaining the resource-releasing operation in the resource exclusion system shown Fig. 1.

Fig. 4 is a flowchart for explaining the resource-holding operation in a second embodiment of the present invention.

Fig. 5 is a flowchart for explaining the resource-releasing operation in the second embodiment of the present invention.

[Explanation of Reference Numerals]

11...resource

12...exclusive control management table

13a, 13b...process

Fig. 1

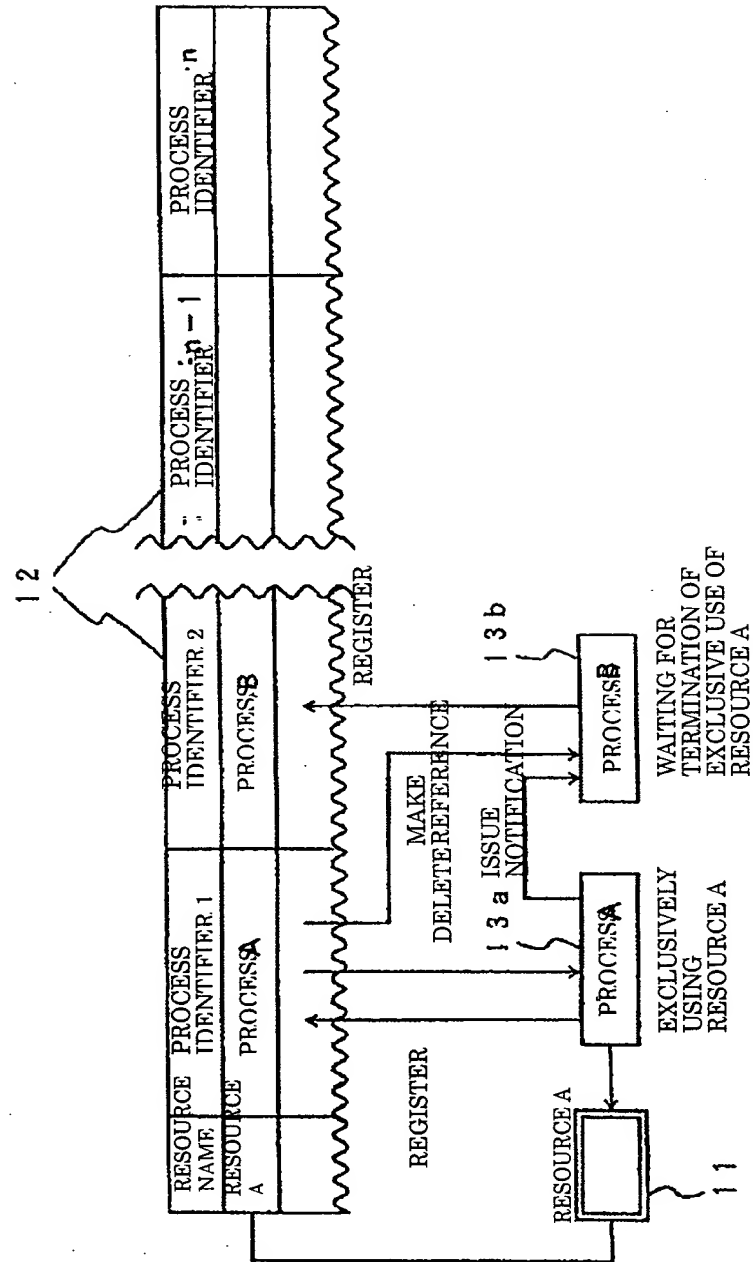


Fig.2

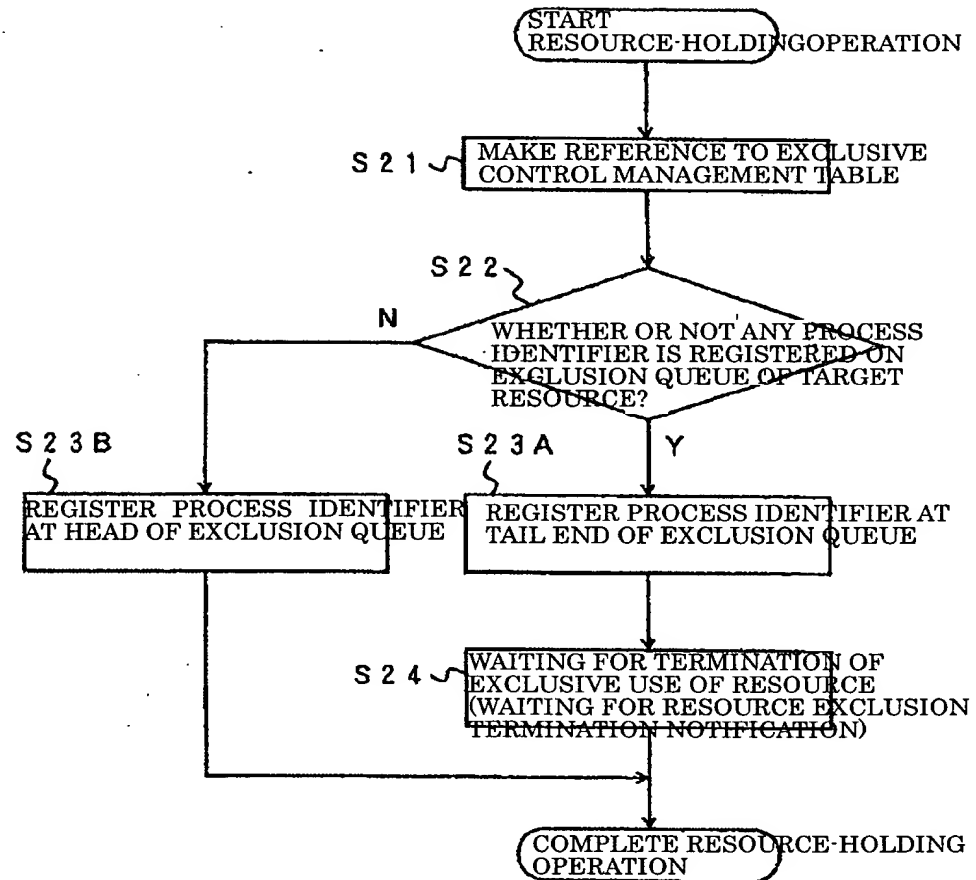


Fig.3

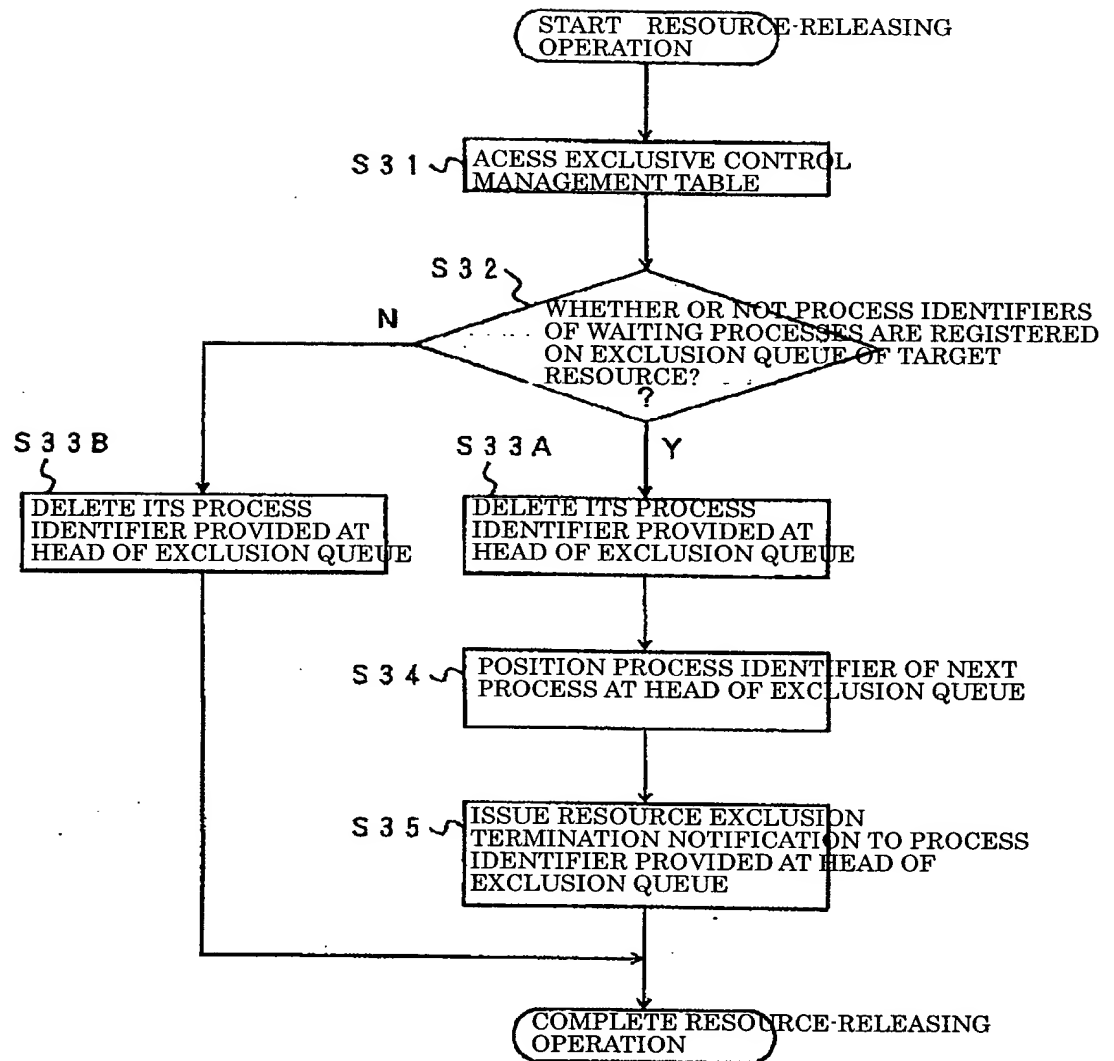


Fig.4

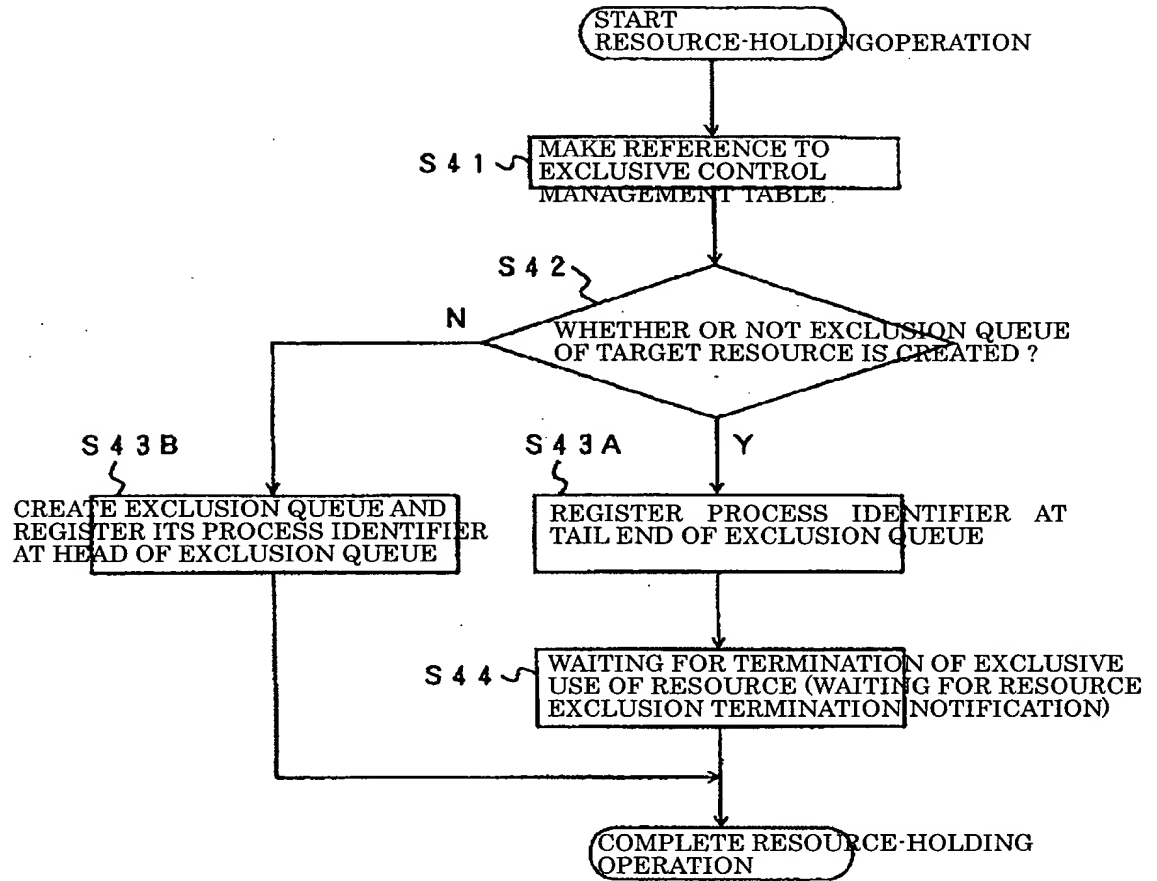


Fig.5

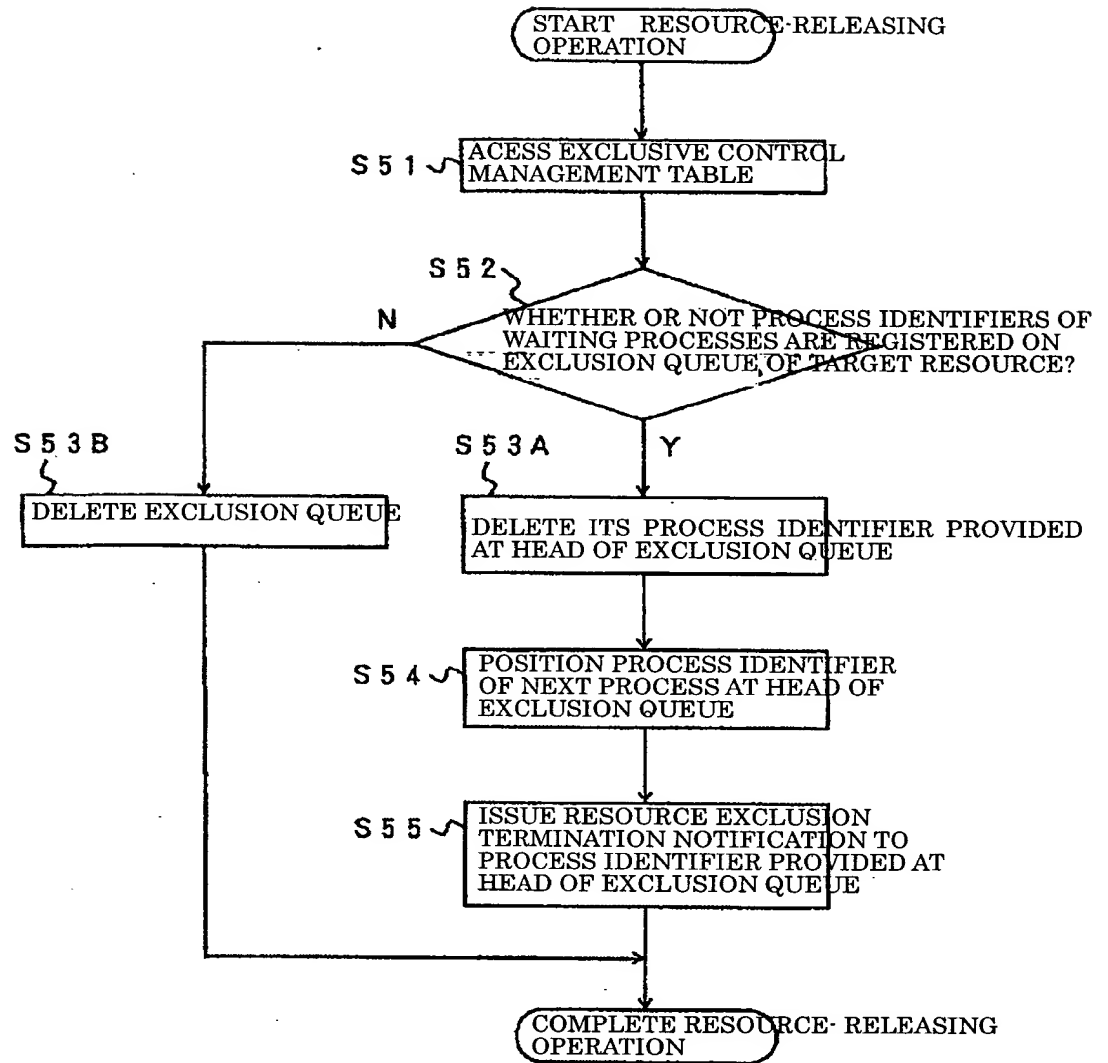


FIG. 1

RESOURCE NAME

PROCESS IDENTIFIER 1

PROCESS IDENTIFIER 2

PROCESS IDENTIFIER $n-1$

PROCESS IDENTIFIER n

RESOURCE A

PROCESS A

PROCESS B

11 RESOURCE A

13a PROCESS A

EXCLUSIVELY USING RESOURCE A

13b PROCESS B

WAITING FOR TERMINATION OF EXCLUSIVE USE OF RESOURCE A

REGISTER

DELETE

ISSUE NOTIFICATION

MAKE REFERENCE

REGISTER

FIG. 2

START RESOURCE-HOLDING OPERATION

S21 MAKE REFERENCE TO EXCLUSIVE CONTROL MANAGEMENT
TABLE

S22 WHETHER OR NOT ANY PROCESS IDENTIFIER IS
REGISTERED ON EXCLUSION QUEUE OF TARGET RESOURCE?

S23A REGISTER PROCESS IDENTIFIER AT TAIL END OF

EXCLUSION QUEUE

S23B REGISTER PROCESS IDENTIFIER AT HEAD OF EXCLUSION QUEUE

S24 WAITING FOR TERMINATION OF EXCLUSIVE USE OF RESOURCE (WAITING FOR RESOURCE EXCLUSION TERMINATION NOTIFICATION)

COMPLETE RESOURCE-HOLDING OPERATION

FIG. 3

START RESOURCE-RELEASING OPERATION

S31 ACCESS EXCLUSIVE CONTROL MANAGEMENT TABLE

S32 WHETHER OR NOT ANY PROCESS IDENTIFIER ON THE WAITING IS REGISTERED ON EXCLUSION QUEUE OF TARGET RESOURCE?

S33A DELETE ITS PROCESS IDENTIFIER PROVIDED AT HEAD OF EXCLUSION QUEUE

S33B DELETE ITS PROCESS IDENTIFIER PROVIDED AT HEAD OF EXCLUSION QUEUE

S34 POSITION PROCESS IDENTIFIER OF NEXT PROCESS AT HEAD OF EXCLUSION QUEUE

S35 ISSUE RESOURCE EXCLUSION TERMINATION NOTIFICATION TO PROCESS IDENTIFIER PROVIDED AT HEAD OF EXCLUSION QUEUE

COMPLETE RESOURCE-RELEASING OPERATION

FIG. 4

START RESOURCE-HOLDING OPERATION

S41 MAKE REFERENCE TO EXCLUSIVE CONTROL MANAGEMENT TABLE

S42 WHETHER OR NOT EXCLUSION QUEUE OF TARGET

RESOURCE IS CREATED ?

S43A REGISTER PROCESS IDENTIFIER AT TAIL END OF
EXCLUSION QUEUE

S43B CREATE EXCLUSION QUEUE AND REGISTER ITS PROCESS
IDENTIFIER AT HEAD OF EXCLUSION QUEUE

S44 WAITING FOR TERMINATION OF EXCLUSIVE USE OF
RESOURCE (WAITING FOR RESOURCE EXCLUSION TERMINATION
NOTIFICATION)

COMPLETE-RESOURCE HOLDING OPERATION

FIG. 5

START RESOURCE-RELEASING OPERATION

S51 ACCESS EXCLUSIVE CONTROL MANAGEMENT TABLE

S32 WHETHER OR NOT ANY PROCESS IDENTIFIER ON THE
WAITING IS REGISTERED ON EXCLUSION QUEUE OF TARGET
RESOURCE?

S53A DELETE ITS PROCESS IDENTIFIER PROVIDED AT HEAD OF
EXCLUSION QUEUE

S53B DELETE EXCLUSION QUEUE

S54 POSITION PROCESS IDENTIFIER OF NEXT PROCESS AT HEAD
OF EXCLUSION QUEUE

S55 ISSUE RESOURCE EXCLUSION TERMINATION NOTIFICATION
TO PROCESS IDENTIFIER PROVIDED AT HEAD OF EXCLUSION
QUEUE

COMPLETE RESOURCE-RELEASING OPERATION

THIS PAGE BLANK (USPTO)

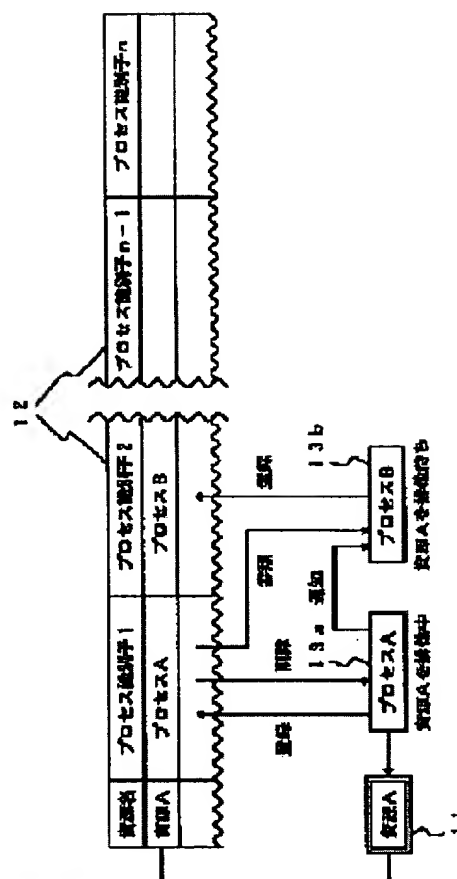
RESOURCE EXCLUSIVE SYSTEM

Patent number: JP10049386
Publication date: 1998-02-20
Inventor: ONO KAZUKI
Applicant: NIPPON ELECTRIC CO
Classification:
 - International: G06F9/46
 - european:
Application number: JP19960204596 19960802
Priority number(s): JP19960204596 19960802

Report a data error here

Abstract of JP10049386

PROBLEM TO BE SOLVED: To provide the resource exclusive system which never causes a decrease in throughput even if the multiplicity of processes increases by decentralizing serial processing needed for exclusive control to respective processes, and eliminating a bottleneck of an exclusive control processing process. **SOLUTION:** Processes 13a and 13b to secure resources refer to and update an exclusive control management table 12 having resource names and a queue. Each process when making a request to secure a resource registers its own identifier at the tail of an exclusive queue when a process identifier is registered in the exclusive queue. When the process identifier is not registered in the exclusive queue, on the other hand, its own identifier is registered at its head to secure the resource. When releasing the resource, the process deletes its own identifier from the exclusive queue. At this time, when the identifier of another process is registered in the exclusive queue, resource exclusion completion is reported.



Data supplied from the esp@cenet database - Worldwide

THIS PAGE BLANK (USPTO)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-49386

(43) 公開日 平成10年(1998) 2月20日

(51) Int.Cl.⁶

G 0 6 F 9/46

識別記号

3 4 0

庁内整理番号

F 1

G 0 6 F 9/46

技術表示箇所

3 4 0 F

審査請求 有 請求項の数 4 O L (全 9 頁)

(21) 出願番号 特願平8-204596

(22) 出願日 平成8年(1996) 8月2日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 小野 一樹

東京都港区芝五丁目7番1号 日本電気株式会社内

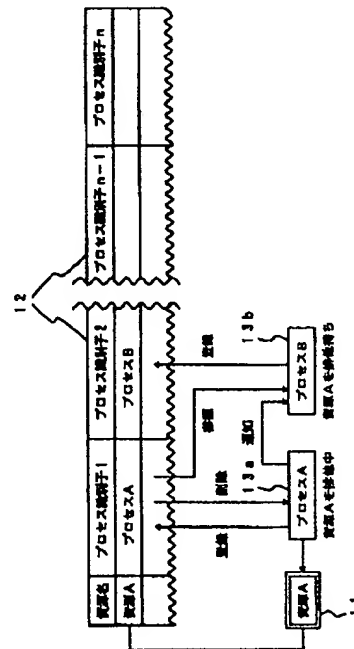
(74) 代理人 弁理士 後藤 洋介 (外2名)

(54) 【発明の名称】 資源排他方式

(57) 【要約】

【課題】 排他制御に必要な一連の処理を各プロセスに分散させ、排他制御処理プロセスでのボトルネックをなくして、プロセスの多重度が増加しても、スループットの低下を招くことがない資源排他方式を提供する。

【解決手段】 資源名と排他待ち行列とを有する排他制御管理テーブル12に対して、資源11を確保しようとするプロセス13a、13bが、それぞれ、参照・更新を行うようにする。各プロセスは、資源確保を要求する場合、排他待ち行列にプロセス識別子が登録されていれば、その最後尾に自プロセスの識別子を登録する。排他待ち行列にプロセス識別子が登録されていなければ、その先頭に自プロセスの識別子を登録し、資源を確保する。資源を解放する場合、各プロセスは、排他待ち行列に登録された自プロセスの識別子を削除する。この際、排他待ち行列に他のプロセスの識別子が登録されている場合は、資源排他完了を通知する。



THIS PAGE BLANK (USPTO)

【特許請求の範囲】

【請求項1】 複数のプロセスが確保を要求し得る資源を、前記複数のプロセスの1つに占有させる資源排他方式において、前記資源の名称と、該資源の名称に対応する排他待ち行列とを有する排他制御管理テーブルに対し、前記複数のプロセスがそれぞれ参照・更新を行うようにしたことを特徴とする資源排他方式。

【請求項2】 前記複数のプロセスは、それぞれ、前記資源を確保しようとするタイミングで、前記排他制御管理テーブルを参照し、前記排他待ち行列に前記複数のプロセスに付与された識別子のいずれかが登録されている場合は、自プロセスの識別子を前記資源待ち行列の最後尾に登録して資源待ちを行い、前記排他待ち行列に前記識別子が登録されていない場合は、当該資源待ち行列の先頭に自プロセスの識別子を登録して前記資源を確保することを特徴とする請求項1の資源排他方式。

【請求項3】 前記複数のプロセスは、それぞれ、確保した資源を解放しようとするタイミングで、前記資源待ち行列に登録した自プロセスの識別子を削除して当該資源を解放し、解放した資源に対応する前記資源待ち行列項目に識別子が登録されている場合に、該識別子に対応するプロセスに対して資源の解放を通知することを特徴とする請求項2の資源排他方式。

【請求項4】 前記排他制御管理テーブルの前記排他待ち行列が、前記複数のプロセスによって、作成され削除されることを特徴とする請求項1、2、または3の資源排他方式。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】 本発明は、資源排他方式に関し、特にコンピュータシステムにおいて複数のプロセス間での資源排他管理を行う資源排他方式に関する。

【0002】

【従来の技術】 従来、複数のプロセス間における資源の排他処理を行う資源排他方式では、排他制御処理を行うプロセスが、資源を要求したプロセスからの「排他要求受付」、「排他管理テーブルへの排他情報の登録」、「資源を要求したプロセスから資源が解放されるまでの資源のロック」、「資源解放要求受付」、「資源解放時の排他管理テーブルからの排他情報の削除」、「資源解放時のロック解除通知」の全ての処理を行っている。この種の資源排他方式としては、例えば、特開平5-113891号公報や、特開平5-20269号公報等に、様々な方法が提案されている。

【0003】

【発明が解決しようとする課題】 第1の問題点は、資源の排他処理を行う排他制御処理プロセスが、システム全体のボトルネックとなっていることである。

【0004】 その理由は、排他制御処理を行うプロセスが、資源を要求する他のプロセス全てに対して排他制御

を行う必要があるからである。つまり、排他制御処理を行うプロセスは、「排他要求受付」、「排他管理テーブルへの排他情報の登録」、「資源を要求したプロセスから資源が解放されるまでの資源のロック」、「資源解放要求受付」、「資源解放時の排他管理テーブルからの排他情報の削除」、「資源解放時のロック解除通知」等の一連の排他制御処理を、資源を要求する他のプロセスのそれぞれについて行わなければならない、排他制御を必要とするプロセスの数が増加するにしたがって、排他制御処理プロセスのオーバーヘッドが増大し、その結果、システム処理のボトルネックとなって、全プロセス（システム全体）のスループット低下を招くからである。

【0005】 本発明は、排他制御に必要な一連の処理を各プロセスに分散させ、排他制御処理プロセスでのボトルネックをなくして、プロセスの多重度が増加しても、スループットの低下を招くことがない資源排他方式を提供することを目的とする。

【0006】

【課題を解決するための手段】 本発明によれば、複数のプロセスが確保を要求し得る資源を、前記複数のプロセスの1つに占有させる資源排他方式において、前記資源の名称と、該資源の名称に対応する排他待ち行列とを有する排他制御管理テーブルに対し、前記複数のプロセスがそれぞれ参照・更新を行うようにしたことを特徴とする資源排他方式がえられる。

【0007】 ここで、前記複数のプロセスは、それぞれ、前記資源を確保しようとするタイミングで、前記排他制御管理テーブルを参照し、前記排他待ち行列に前記複数のプロセスに付与された識別子のいずれかが登録されている場合は、自プロセスの識別子を前記資源待ち行列の最後尾に登録して資源待ちを行い、前記排他待ち行列に前記識別子が登録されていない場合は、当該資源待ち行列の先頭に自プロセスの識別子を登録して前記資源を確保する。

【0008】 また、前記複数のプロセスは、それぞれ、確保した資源を解放しようとするタイミングで、前記資源待ち行列に登録した自プロセスの識別子を削除して当該資源を解放し、解放した資源に対応する前記資源待ち行列項目に識別子が登録されている場合に、該識別子に対応するプロセスに対して資源の解放を通知する。

【0009】

【作用】 排他制御管理テーブルは、排他対象とする資源の資源名とそれに対応する排他待ち行列を有する。プロセスは、資源を確保するために、排他制御管理テーブルを参照し、自プロセスが確保しようとしている資源に対して他のプロセスの識別子が登録されていないことを判別して、自プロセスの識別子を排他制御管理テーブルの待ち行列へ登録して資源の確保を行う。

【0010】 確保の対象である資源の排他待ち行列に、自プロセス以外の識別子が登録されている場合には、待

THIS PAGE BLANK (USPTO)

ち行列の最後尾に自プロセスの識別子を登録して排他待ちとする。

【0011】資源解放処理を行う場合は、排他制御管理テーブル上の対象資源の排他待ち行列から自プロセスの識別子を削除する。このとき、排他待ち行列に他のプロセスの識別子が登録されている場合には、そのプロセスに対して排他完了の通知を行う。

【0012】

【発明の実施の形態】以下に図面を参照して本発明の実施の形態について説明する。

【0013】図1に本発明の資源排他方式の第1の実施の形態を示す。この資源排他方式は、資源11を特定するための資源名と、各資源11に対応する待ち行列と、を有する排他制御管理テーブル12と、排他制御管理テーブル12に対してそれぞれ参照・更新を行うことができるプロセス13a、13bを含む。なお、排他制御管理テーブル12は、高速アクセス可能な装置上に作成されることが望ましい。

【0014】以下、図2及び図3をも参照して図1の資源排他方式の動作を説明する。

【0015】まず、図2を参照して、資源確保時の処理について説明する。ここでは、排他制御管理テーブル12に、資源11の資源名として「資源A」が登録されており、資源Aに対応する待ち行列には、識別子が全く登録されていないとする。このような状況において、プロセス13aが、資源11を確保しようとする場合、プロセス13aは、まず、ステップS21において、排他制御管理テーブル12を参照する。そして、ステップS22において、対象資源（この場合資源A）の排他待ち行列上に識別子が登録されていないと判定する。そして、プロセス13aは、ステップS23Bにおいて、自プロセスのプロセス識別子「プロセスA」を対象資源の排他待ち行列の先頭（プロセス識別子1）に登録して、対象資源11を確保する。

【0016】この後、プロセス13bが、資源11を確保しようとする場合は、ステップS22において、対象資源11の排他待ち行列上にプロセス識別子が登録されていると判定されるので、ステップS23Aにおいて、対象資源11の排他待ち行列の最後尾（この場合、プロセス識別子2）に自プロセスのプロセス識別子「プロセスB」を登録する。そして、プロセス13bは、ステップS24で、排他待ち状態に移して、対象資源11を確保しているプロセス13aからの資源排他完了通知を待つ。

【0017】次に、図3を参照して、資源解放時の処理について説明する。ここで、排他制御管理テーブル12は、図1の状態、即ち、プロセス13aが資源11を確保しており、プロセス13bが排他待ちの状態にあるとする。

【0018】まず、プロセス13aは、ステップS31

の排他制御管理テーブル12の、確保していた資源11の排他待ち行列を参照する。そして、ステップS32において、対象資源の排他待ち行列上に待ちのプロセス識別子が登録されているか判断する。図1の場合、プロセス13bのプロセス識別子が登録されているので、ステップS33Aで、自プロセスのプロセス識別子を削除して、待ち状態にあるプロセス識別子を先頭の方へ1段スライドさせる。この結果、プロセス識別子「プロセスB」は、プロセス識別子1の欄に登録させる。そして、プロセス13aは、排他待ち行列の先頭に登録されているプロセス識別子に対応するプロセス13bに対して、資源排他完了通知を発行する。そして、資源排他完了通知を受け取ったプロセス13bは、資源11を確保する。

【0019】このあと、プロセス13bが、資源11を解放する場合には、ステップS32において、対象資源の排他待ち行列上に待ちのプロセス識別子が登録されていないと判断されるので、ステップS33Bにおいて、自プロセスのプロセス識別子を削除して、資源解放処理を終える。

【0020】次に、図4及び図5を参照して本発明の第2の実施の形態について説明する。本実施の形態においては、排他制御管理テーブルには待ち行列が形成されておらず、各プロセスが、必要に応じて、排他制御管理テーブルに待ち行列を作成する。

【0021】資源確保時の処理は、図4に示すように、資源を確保しようとするプロセスが、まず、ステップS41において、排他制御管理テーブルを参照する。そしてそのプロセスは、ステップS42において、排他制御管理テーブルに、対象資源の排他待ち行列が作成されているか判定する。排他制御管理テーブルに、対象資源の排他待ち行列が作成されている場合は、対象資源が他のプロセスに確保されているので（排他中なので）、ステップS43Aにおいて、対象資源の排他待ち行列の最後尾に自プロセスのプロセス識別子を登録する。それから、そのプロセスは、ステップS44で、排他待ち状態に移して、対象資源を確保している先行プロセスからの資源排他完了通知を待つ。

【0022】ステップS42において、排他制御管理テーブルに、対象資源の排他待ち行列が作成されていない場合は、そのプロセスは、対象資源の排他が可能（確保が可能）なので、ステップS43Bで、対象資源の排他待ち行列を作成し、その先頭に自プロセスのプロセス識別子を登録する。これにより、そのプロセスは、対象資源を確保する。

【0023】次に、図5を参照して、資源解放時の処理について説明する。まず、資源解放を行うプロセスは、ステップS51で、自プロセスが確保していた資源の排他待ち行列を参照する。そして、ステップS52で、対象資源の排他待ち行列上に他のプロセス識別子が、登録

THIS PAGE BLANK (USPTO)

されているか判断する。対象資源の排他待ち行列上に他のプロセス識別子が登録されている場合、対象資源の排他待ちプロセスが存在するので、ステップS53Aで、対象資源の排他待ち行列から自プロセスのプロセス識別子を削除して、ステップS54で、対象資源の排他待ち行列上の他のプロセス識別子を1段先頭側へスライドさせる。そして、その対象資源の排他待ち行列の先頭に位置する他のプロセス識別子に対応するプロセスに対して資源排他完了通知を発行する。これにより、排他待ちを行っていたプロセスは、資源の使用が可能となる。

【0024】ステップS52で、対象資源の排他待ち行列上に他のプロセス識別子が登録されていない場合は、対象資源の排他待ちを行っているプロセスが存在しないので、自プロセスのプロセス識別子を削除するとともに、排他制御管理テーブル上の、解放使用とする資源の排他待ち行列を削除する。

【0025】

【発明の効果】第1の効果は、プロセス多重度が増加した場合であっても、排他制御処理プロセスのボトルネックによるスループットの低下を回避することができる。

【0026】その理由は、「排他要求受付」、「排他管

理テーブルへの排他情報の登録」、「資源を要求したプロセスから資源が解放されるまでの資源のロック」、「資源解放要求受付」、「資源解放時の排他管理テーブルからの排他情報の削除」、「資源解放時のロック解除通知」等の一連の排他制御処理を、資源を使用する各プロセスに分散実行させるようにしたからである。

【図面の簡単な説明】

【図1】本発明の第1の実施の形態を示すブロック図である。

【図2】図1の資源排他方式の資源確保処理を説明するためのフローチャートである。

【図3】図1の資源排他方式の資源解放処理を説明するためのフローチャートである。

【図4】本発明の第2の実施の形態の資源確保処理を説明するためのフローチャートである。

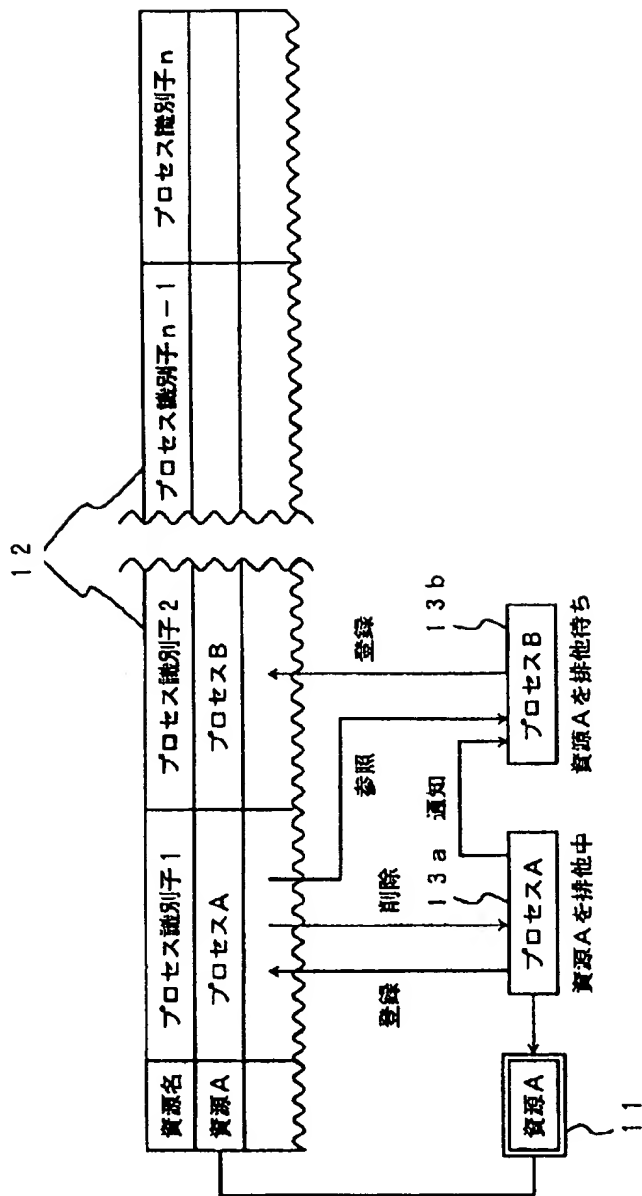
【図5】本発明の第2の実施の形態の資源解放処理を説明するためのフローチャートである。

【符号の説明】

- 11 資源
- 12 排他制御管理テーブル
- 13a, 13b プロセス

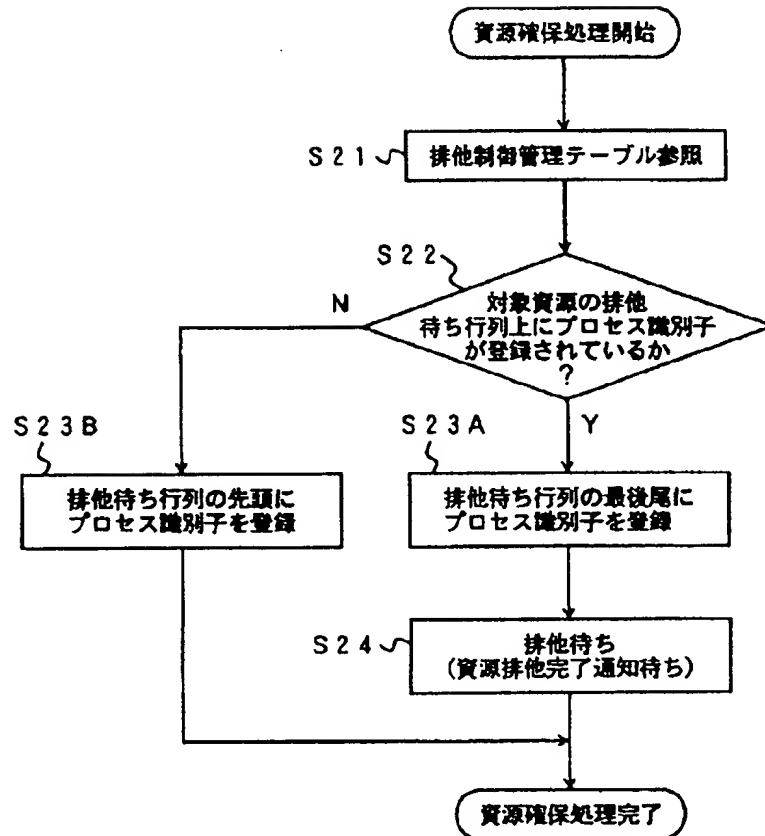
THIS PAGE BLANK (USPTO)

【図1】



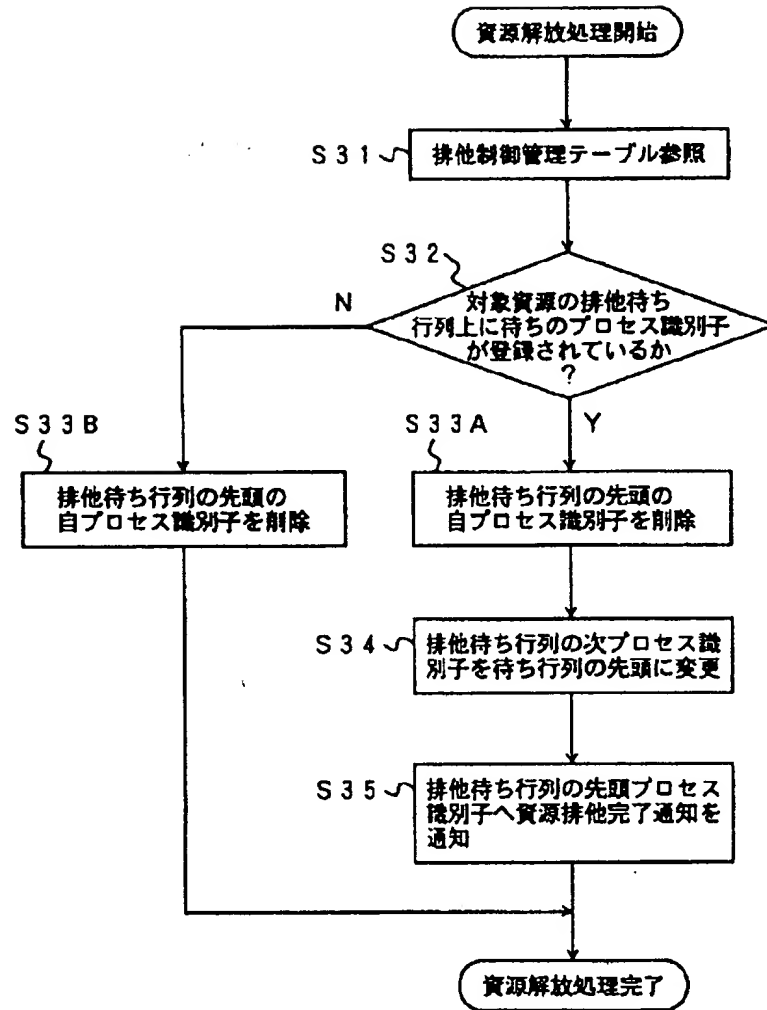
THIS PAGE BLANK (USPTO)

【図2】



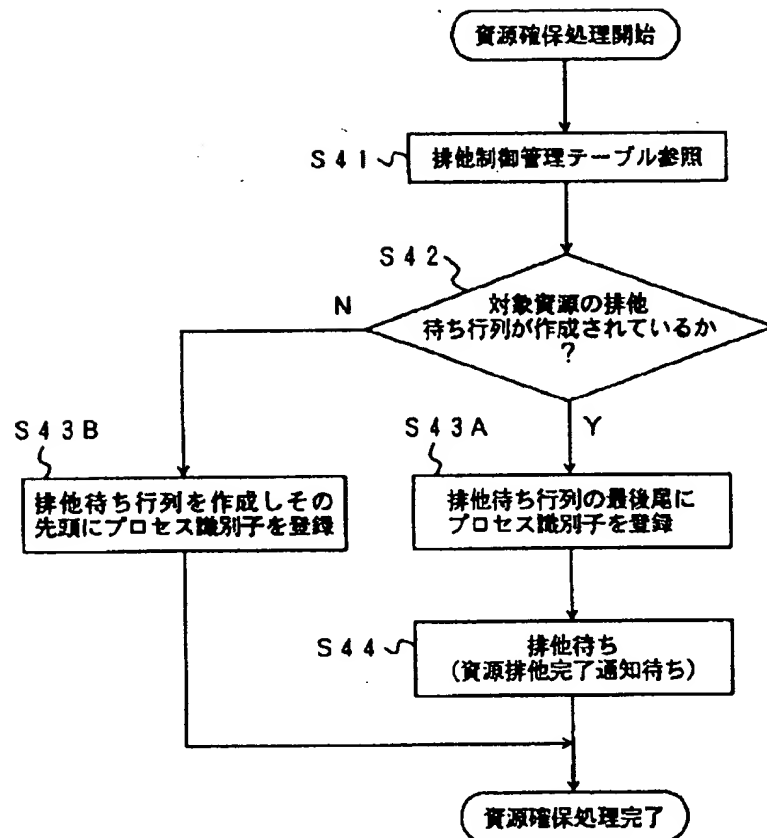
THIS PAGE BLANK (USPTO)

【図3】



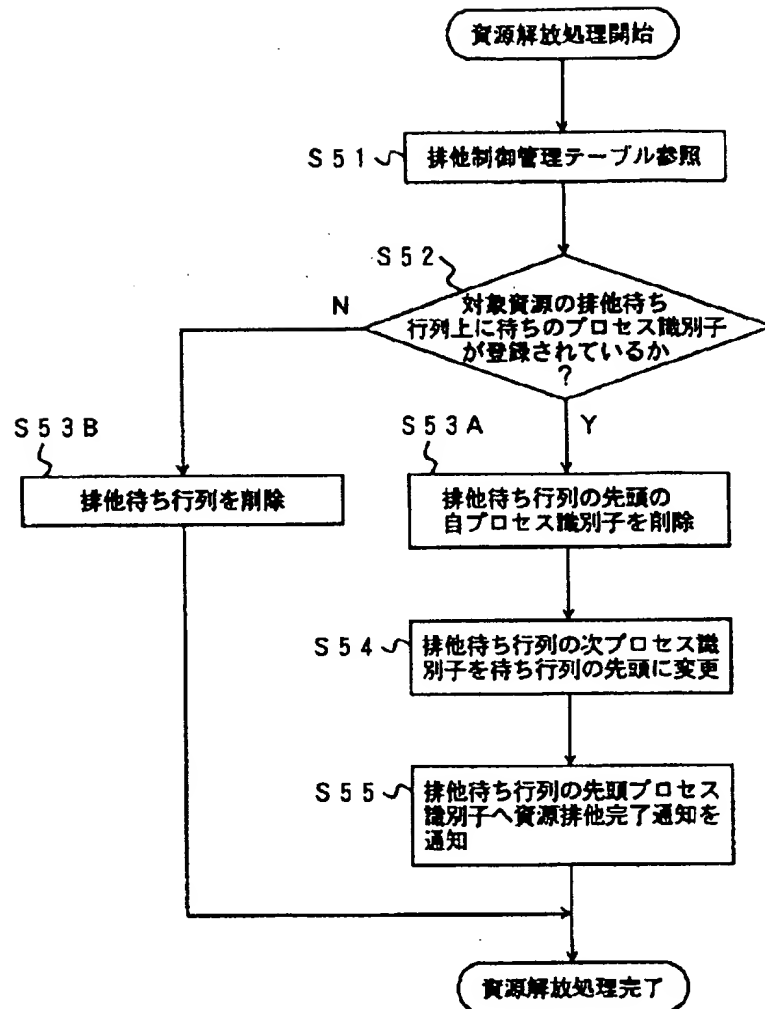
THIS PAGE BLANK (USPTO)

【図4】



THIS PAGE BLANK (USPTO)

【図5】



THIS PAGE BLANK (USPTO)